Neural Collapse is Globally Optimal in Deep Regularized ResNets and Transformers

Author names withheld

Under Review for the Workshop on High-dimensional Learning Dynamics, 2025

Abstract

The empirical emergence of neural collapse—a surprising symmetry in the feature representations of the training data in the penultimate layer of deep neural networks—has spurred a line of theoretical research aimed at its understanding. However, existing work either focuses on data-agnostic models or it remains limited to multi-layer perceptrons. We fill both these gaps by analyzing modern architectures in a data-aware regime: we prove that global optima of deep regularized transformers and residual networks (ResNets) with LayerNorm trained with cross entropy or mean squared error loss are approximately collapsed, and the approximation gets tighter as the depth grows. More generally, we formally reduce any end-to-end large-depth ResNet or transformer training into an equivalent unconstrained features model, thus justifying its wide use in the literature even beyond data-agnostic settings. Our theoretical results are supported by experiments on computer vision and language datasets showing that, as the depth grows, neural collapse indeed becomes more prominent.

1. Introduction

In 2020, Papyan et al. [40] discovered a surprising geometric structure in learned representations of deep neural networks (DNNs) at convergence dubbed "neural collapse" (NC). It concerns the representations of the training samples in the last layer of the network: the feature vectors of the samples from the same class converge to the respective class-mean (NC1); the class-means form a simplex equiangular tight frame (ETF), maximizing the pairwise angles (NC2); finally, the class-means align with the rows of the weight matrix of the last layer (NC3). Similar structures were also subsequently discovered for class-imbalanced classification [48], regression [1] and language modeling [59], demonstrating that neural collapse is ubiquitous when training deep models.

To understand the emergence of NC, Mixon et al. [39] introduced the "unconstrained features model" (UFM). In the UFM, one assumes the features of the last layer to be free variables and optimizes over them together with the weight matrix of the last layer. Using the UFM, the optimality of the NC has been proven, as well as its emergence during gradient descent training in various settings, see Section 2. However, the UFM has since been criticized [22] for being too simplistic and data-agnostic. However, UFM-free results so far cover shallow (up to three layers) networks [19, 26, 58] or come with strong assumptions [3, 22, 41, 43, 55, 61] (see Section 2). Moreover, with the exception of [55], all works only focus on multi-layer perceptrons (MLPs). However, NC is equally relevant in modern architectures, such as ResNets ([17]) or transformers ([51]) [55, 59].

In this work, we fill both mentioned gaps at once. First, we are the first to theoretically analyze NC in ResNets with LayerNorm and transformers. Second, our results prove end-to-end approximate

optimality of NC in training with weight regularization. This has only ever been done for MLPs with deep linear heads in [22]. To be more precise, our contributions are summarized below.

For ResNets and transformers with one linear layer per MLP block and constant regularization strength, we prove that NC is the asymptotically optimal solution as the number of blocks goes to infinity. All global optima in deep-enough networks must be approximately collapsed and the distance from perfect collapse is non-asymptotically upper-bounded in terms of the depth. These results hold for both CE and MSE losses, under minimal assumptions on the data. We support these findings by experiments on computer vision datasets with ResNets and vision transformers, showing that the amount of collapse increases with the depth of the architecture, as predicted by our theory.

More generally, we provide a formal connection between deep ResNets/transformers and unconstrained features models: we prove that, as these architectures become deeper, their global optima converge to those of an equivalent UFM. This result holds for a wide class of continuous losses and justifies the use of UFM for the analysis of NC.

2. Related work

Related work on the UFM is reviewed in Appendix A. In the end-to-end training regime, conditions on data that make NC feasible in the shallow case are identified in [19]. Two-layer networks are considered in [26], which uses NTK theory and other kernel methods to conclude that NC in this regime is rather restricted. To the contrary, in the mean-field regime, positive results about NC1 are given in [58] for certain three-layer networks. In the deep case, convergence to NC is studied in [22, 41, 43, 61]. However, a block-structured empirical NTK is assumed in [43], and symmetric quasi-interpolation is required in [41, 61]. The former does not justify this assumption, while the latter requires an unusual weight regularization and interpolators with a given norm. Wide networks are considered in [22], which proves the emergence of NC1 requiring at least the last two layers to be linear (and even deeper linear heads for NC2 and NC3).

Closer to the scope of the current work, NC is studied in ResNets in [55]. Two main claims are proved: the monotonicity of NC1-NC2 metrics across layers of ResNets, and a negative result about collapse in a variant of UFM similar to the one considered in [50]. However, the monotonicity is proved under the strong assumption that the data evolves across layers on a geodesic, which is not possible in general since one can construct configurations where samples from different classes would collide. Moreover, the UFM taken into account is based on a heuristic derivation (a link between representation cost and transport cost of the features) that does not hold exactly in practice.

3. Preliminaries

Notation. We study two different data formats and architectures. For ResNets, the input data and one-hot labels are $X_0 \in \mathbb{R}^{d_0 \times N}$ and $Y \in \mathbb{R}^{K \times N}$, where d_0 is the input dimension, N the number of samples and K the number of classes; whereas for transformers it is $X_0 \in \mathbb{R}^{N \times V \times C}$ and $Y \in \mathbb{R}^{N \times K \times C}$, where C is the context length (number of tokens in the prompt) and V the vocabulary size (number of distinct tokens). We use x_{ki} to indicate the i-th sample of the k-th class. For transformers, a sample corresponds to the position of each individual token and, thus, x_{ki} corresponds to a token position labeled as class k, with samples ordered arbitrarily. We assume a class-balanced setting with n samples per class.

ResNets and transformers. Let $\mathrm{id}(\cdot)$ be the identity mapping, σ be the ReLU and LN the LayerNorm that subtracts the mean of each vector of the last dimension of the input tensor by its own mean and then normalizes it by its own standard deviation.

Definition 1 *An L-block ResNet with LayerNorm and one linear layer per block (L-RN1), and an L-block transformer with one linear layer per attention sub-block and MLP sub-block (L-T11) with intermediate dimension d are defined as*

$$f_\theta(Z) = \mathrm{lin}_L \circ \mathrm{LN}_L \circ \mathrm{B}_{L-1} \circ \cdots \circ \mathrm{B}_1 \circ \mathrm{Embed}(Z). \tag{1}$$

Here, Z is a vector for the ResNet and a matrix for the transformer, and $\mathrm{lin}_L(Z) = W_L Z$ is the last layer; $\mathrm{Embed}(Z) = W_e Z + W_p$ ($\mathrm{Embed}(Z) = W_0 X_0 + b_0$) is the embedding layer of the transformer (ResNet) with W_e being the token embedding and W_p the positional embedding (W_0, b_0 being the embedding linear layer). For the transformer, $\mathrm{B}_l = \mathrm{MLP}_l \circ \mathrm{LN}_{l,2} \circ \mathrm{ATTN}_l \circ \mathrm{LN}_{l,1}$. Such block consists of the normalization layers $\mathrm{LN}_{l,1}, \mathrm{LN}_{l,2}$, the MLP $\mathrm{MLP}_l(Z) = Z + \sigma(W_l Z + b_l)$ and the single-head attention (the matrix M is the causal mask)

$$\mathrm{ATTN}_l(Z) = Z + W_{VO} Z A_l(Z),\ A_l(Z) = \mathrm{softmax}(M + Z^T W_{QK} Z/\sqrt{d}). \tag{2}$$

For the ResNet, $\mathrm{B}_l = (\mathrm{id} + \sigma \circ \mathrm{lin}_l) \circ \mathrm{LN}$. Denote by θ the collection of all learnable parameters. Denote $X_1 = \mathrm{Embed}(X_0)$; $X_{l+1} = \mathrm{B}_l(X_l)$ for $l \in \{1, \ldots, L-1\}$; $f_\theta(X_0) = X_{L+1} := W_L \,\mathrm{LN}(X_L)$ the intermediate representations of the training data stored in a matrix (or tensor) form.

Neural collapse metrics and generalized unconstrained features model (GUFM). Let $h_\theta(\cdot)$ be the output of the penultimate layer. We denote by x_{ki} the i-th sample of the k-th class. We define $\mu_k := \frac{1}{n}\sum_{i=1}^n x_{ki}$ as the class-means in the l-th layer and $\mu_G := \frac{1}{K}\sum_{k=1}^K \mu_k$ as the global mean. Let $\Sigma_W := \frac{1}{N}\sum_{k,i=1}^{K,n}(x_{k,i} - \mu_k)(x_{k,i} - \mu_k)^T$ and $\Sigma_B := \frac{1}{K}\sum_{k=1}^K(\mu_k - \mu_G)(\mu_k - \mu_G)^T$ be the within- and between-class variability matrices, and let $E_K = I_K - \mathbf{1}_K \mathbf{1}_K^T$ be the un-rotated ETF matrix.

Definition 2 *Any pair (W, X) of matrices s.t. W has at least as many columns as rows, X has $N = Kn$ columns and they can multiply as WX has the following NC metrics:*

- $\mathrm{NC1}(W, X) = \frac{\mathrm{tr}(\Sigma_W)}{\mathrm{tr}(\Sigma_B)}$, *i.e., the ratio of within- and between-class variability.*
- $\mathrm{NC2A}(W, X) = \frac{\min_{c\geq 0}\|WW^T - cE_K\|_F}{\|WW^T\|_F}$, *i.e. the distance of WW^T from the closest (scaled) ETF.*
- $\mathrm{NC2B}(W, X) = \frac{\min_{c\geq 0}\|WW^T - cI_K\|_F}{\|WW^T\|_F}$, *i.e. the distance of WW^T from the closest (scaled) identity.*
- $\mathrm{NC3}(W, X) = 1 - \frac{1}{N}\sum_{k,i=1}^{K,n}\cos(x_{ki}, W_{k:})$, *i.e., one minus the average cosine similarity between the samples and the corresponding row of W.*

A model is said to exhibit NC if all metrics are 0 and approximate NC if all metrics are close to zero. NC2A is defined for CE loss or MSE loss with unregularized bias in the last layer, and NC2B is defined for MSE loss with bias-free last layer. We consider the following optimization problem:

$$\min_\theta \ \mathcal{L}(f_\theta(X), Y) + \frac{\lambda}{2}\left\|\bar{\theta}\right\|^2, \tag{3}$$

where $\lambda > 0$, $\bar{\theta}$ is the subset of parameters that excludes biases and the parameters in embedding layers (W_e, W_p, W_0), and $\mathcal{L} \geq 0$ is a continuous loss. Let $\mathcal{L}_{\text{CE}}, \mathcal{L}_{\text{MSE}}$ be CE and MSE loss, and $\mathcal{L}_L(\theta)$ be the loss of the L-RN1 or L-T11 architecture (depending on the context) with parameters θ. We denote by $\mathcal{L}_L^*$ the optimal such loss value and by $\mathcal{M}_\epsilon^L := \{\theta : \mathcal{L}_L(\theta) \leq \mathcal{L}_L^* + \epsilon\}$ the set of parameters ϵ-close to the optimum. We denote by $\tilde{\mathcal{M}}_L$ the set of all pairs $(W_L, h_\theta(X))$ s.t. θ (including W_L) is in $\mathcal{M}_0^L$. In our analysis, we reduce the end-to-end problem (3) into a simpler UFM:

Definition 3 *Given a continuous loss $\mathcal{L} \geq 0$ and an equivalence relation $\mathcal{R}$ on $\{1, \ldots, N\}$, the generalized unconstrained features model (GUFM) refers to the following optimization problem:*

$$\begin{aligned} \min_{W,X} \ & \mathcal{L}(WX, Y) + \frac{\lambda}{2} \|W\|_F^2, \\ \text{s.t.} \ & \|x_i\| = \sqrt{d}, \quad x_i^T \mathbf{1}_d = 0, \quad \text{for } i \in \{1, \ldots, N\}, \\ & x_i = x_j, \quad \text{for } i, j \in \{1, \ldots, N\}, \quad i \sim_{\mathcal{R}} j, \end{aligned} \tag{4}$$

where $W \in \mathbb{R}^{K \times d}, X = [x_1, \ldots, x_N] \in \mathbb{R}^{d \times N}$ and $Y \in \mathbb{R}^{K \times N}$. Let $\mathcal{L}_{GUFM}(W, X)$ be the loss of the feasible pair (W, X) under this model, $\mathcal{L}_{GUFM}^*$ the optimal such loss and $\mathcal{M}_\epsilon^{GUFM} := \{(W, X) \in \mathcal{M} : \mathcal{L}_{GUFM}(W, X) \leq \mathcal{L}_{GUFM}^* + \epsilon\}$, with $\mathcal{M}$ the set of feasible solutions.

The constraint $x_i^T \mathbf{1}_d = 0$ is due to the LayerNorm just before the last layer in ResNets and transformers. This is w.l.o.g. for CE/MSE loss, for which the optimum is zero-mean. The equivalence relation accounts for potential hard constraints from the input data where we may have identical samples or contexts that may or may not be in the same class. Again, for CE/MSE loss this is w.l.o.g., given that all identical contexts are always labeled with the same class.

4. Main results

Denote as $\text{distmax}(A, B) = \sup_{x \in A} \text{dist}(x, B)$ for any sets A, B.

Theorem 4 *Let the architecture be L-RN1 or L-T11. Assume the inner dimension of the L-T11 (L-RN1) is at least $2V + 2$ (4). Consider the optimization problem (3) with λ independent of the number of layers and its corresponding GUFM (4) with the same loss $\mathcal{L}$ and the equivalence relation defined by pairs of samples (contexts for transformers) in X that coincide. If $\mathcal{L}_{GUFM}^* > 0$, then*

$$\limsup_{L \to \infty} \ \text{distmax}(\tilde{\mathcal{M}}_{\text{L}} \backslash \mathcal{M}_0^{GUFM}, \mathcal{M}_0^{GUFM}) = 0. \tag{5}$$

This result provides a reduction of the end-to-end training objective of a deep-enough architecture to a GUFM using the same loss. This has two important implications. First, it shows that optimal deep ResNets and transformers can represent the optimal solution of the corresponding GUFM problem. As formalized in Corollary 5, this gives a precise characterization of the structure of feature representations in the last layer at the global optimum – the first result of this sort for modern architectures beyond MLPs. Second, it provides a theoretical justification for using the UFM to explain the emergence of NC, showing that the UFM does not oversimplify the problem even when dealing with ResNets and transformers. The proof constructs a sequence of candidate solutions whose representations converge to NC, while the regularization of intermediate layers converges to zero. Global optima must thus converge to represent the optimal GUFM loss and thus also optimal

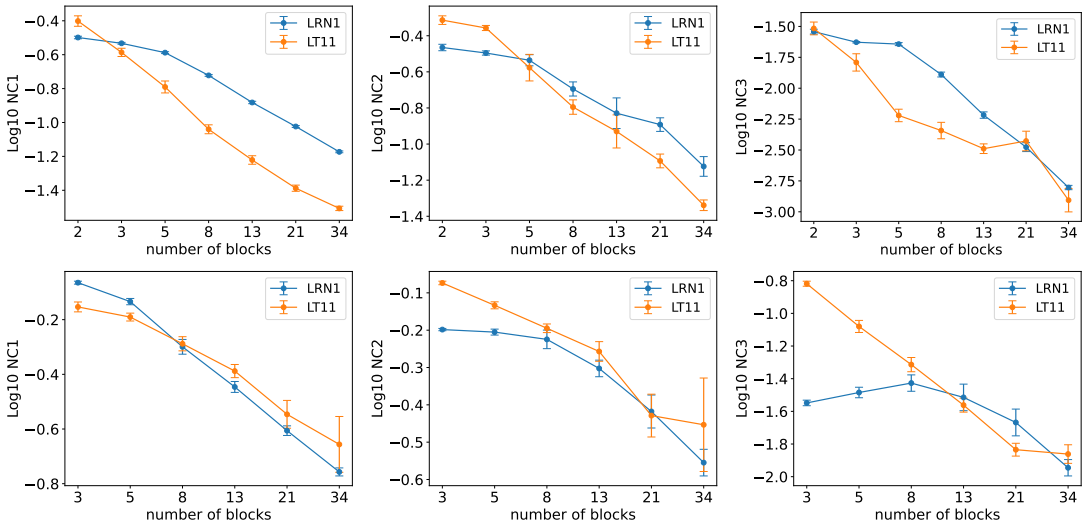

Figure 1: $\log_{10}$ of NC1, NC2 and NC3 metrics respectively in the left, middle and right column, as a function of L for L-RN1, $L-$T11. *Top:* MNIST; *Bottom:* CIFAR10.

GUFM solutions. The complete argument is in Appendix F, with additional notation deferred to Appendix B. We highlight that Theorem 4 holds for any continuous loss. By considering CE or MSE for which the global optima of the corresponding GUFMs are collapsed by Lemma 9 in Appendix C, the emergence of collapse in ResNets and transformers is readily obtained.

Corollary 5 *Let the architecture be L-RN1 or L-T11. For L-RN1 assume all training samples in X to be unique. For L-T11, assume the labels Y to be uniquely determined by the context, i.e., two identical contexts in two different input sequences will be assigned the same label. Using CE or MSE loss, all global optima of the optimization problem (3) exhibit approximate neural collapse which gets tighter as L increases.*

We discuss the non-asymptotic rate of convergence, a language modeling setting and deep neural collapse in Appendix D. Our strategy can be applied to a rather general class of architectures: we consider architectures with two linear layers per MLP block in Appendix E, as well as pre-LN type ResNets, pre-LN type transformers and vision transformers in Appendix G.

5. Experimental results

Our theoretical results suggest an improvement of the NC metrics at the global optima as the depth increases. To empirically verify this effect at moderate depths and for solutions found by SGD, we train ResNets and transformers on MNIST [28] and CIFAR10 [30]. Figure 1 shows the three NC metrics at convergence, as a function of the depth of the architecture. The results are in agreement with the theory developed in Section 4: across different architectures, NC metrics improve with depth, even when the solutions are obtained via SGD. Furthermore, for large-enough depth, the plots roughly follow a log-linear trend, especially for ResNets. This suggests a polynomial dependence between NC metrics and depth L, which is also consistent with our theory, see the remark on the rate of convergence in Appendix D.

References

[1] George Andriopoulos, Zixuan Dong, Li Guo, Zifan Zhao, and Keith Ross. The prevalence of neural collapse in neural multivariate regression. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2024.

[2] Alessio Ansuini, Alessandro Laio, Jakob H Macke, and Davide Zoccolan. Intrinsic dimension of data representations in deep neural networks. *Conference on Neural Information Processing Systems (NeurIPS)*, 2019.

[3] Daniel Beaglehole, Peter Súkeník, Marco Mondelli, and Mikhail Belkin. Average gradient outer product as a mechanism for deep neural collapse. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2024.

[4] Yang Cao, Yanbo Chen, and Weiwei Liu. Prevalence of simplex compression in adversarial deep neural networks. *Proceedings of the National Academy of Sciences*, 122(17):e2421593122, 2025.

[5] Mayee Chen, Daniel Y Fu, Avanika Narayan, Michael Zhang, Zhao Song, Kayvon Fatahalian, and Christopher Ré. Perfectly balanced: Improving transfer and robustness of supervised contrastive learning. In *International Conference on Machine Learning (ICML)*, 2022.

[6] Hien Dang, Tan Nguyen, Tho Tran, Hung Tran, and Nhat Ho. Neural collapse in deep linear network: From balanced to imbalanced data. In *International Conference on Machine Learning (ICML)*, 2023.

[7] Hien Dang, Tho Tran Huu, Tan Minh Nguyen, and Nhat Ho. Neural collapse for cross-entropy class-imbalanced learning with unconstrained relu features model. In *International Conference on Machine Learning (ICML)*, 2024.

[8] Yann Dubois, Stefano Ermon, Tatsunori B Hashimoto, and Percy S Liang. Improving self-supervised learning by characterizing idealized representations. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2022.

[9] Cong Fang, Hangfeng He, Qi Long, and Weijie J Su. Exploring deep neural networks via layer-peeled model: Minority collapse in imbalanced training. In *Proceedings of the National Academy of Sciences (PNAS)*, volume 118, 2021.

[10] Tomer Galanti, Liane Galanti, and Ido Ben-Shaul. On the implicit bias towards minimal depth of deep neural networks. *arXiv preprint arXiv:2202.09028*, 2022.

[11] Tomer Galanti, András György, and Marcus Hutter. Improved generalization bounds for transfer learning via neural collapse. In *First Workshop on Pre-training: Perspectives, Pitfalls, and Paths Forward at ICML*, 2022.

[12] Connall Garrod and Jonathan P Keating. The persistence of neural collapse despite low-rank bias: An analytic perspective through unconstrained features. *arXiv preprint arXiv:2410.23169*, 2024.

[13] Connall Garrod and Jonathan P Keating. Unifying low dimensional observations in deep learning through the deep linear unconstrained feature model. *arXiv preprint arXiv:2404.06106*, 2024.

[14] Jarrod Haas, William Yolland, and Bernhard T Rabus. Linking neural collapse and l2 normalization with improved out-of-distribution detection in deep neural networks. *Transactions on Machine Learning Research (TMLR)*, 2022.

[15] X. Y. Han, Vardan Papyan, and David L Donoho. Neural collapse under mse loss: Proximity to and dynamics on the central path. In *International Conference on Learning Representations (ICLR)*, 2022.

[16] Hangfeng He and Weijie J Su. A law of data separation in deep learning. *Proceedings of the National Academy of Sciences*, 120(36), 2023.

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[18] Wanli Hong and Shuyang Ling. Neural collapse for unconstrained feature model under cross-entropy loss with imbalanced data. *Journal of Machine Learning Research*, 25(192):1–48, 2024.

[19] Wanli Hong and Shuyang Ling. Beyond unconstrained features: Neural collapse for shallow neural networks with general data. *arXiv preprint arXiv:2409.01832*, 2024.

[20] https://www.kaggle.com/datasets/lakshmi25npathi/imdb-dataset-of-50k-movie reviews.

[21] Like Hui, Mikhail Belkin, and Preetum Nakkiran. Limitations of neural collapse for understanding generalization in deep learning. *arXiv preprint arXiv:2202.08384*, 2022.

[22] Arthur Jacot, Peter Súkeník, Zihan Wang, and Marco Mondelli. Wide neural networks trained with weight decay provably exhibit neural collapse. In *International Conference on Learning Representations (ICLR)*, 2025.

[23] Wenlong Ji, Yiping Lu, Yiliang Zhang, Zhun Deng, and Weijie J Su. An unconstrained layer-peeled perspective on neural collapse. In *International Conference on Learning Representations (ICLR)*, 2022.

[24] Jiachen Jiang, Jinxin Zhou, Peng Wang, Qing Qu, Dustin G Mixon, Chong You, and Zhihui Zhu. Generalized neural collapse for a large number of classes. In *Conference on Parsimony and Learning (Recent Spotlight Track)*, 2023.

[25] Kenji Kawaguchi, Zhun Deng, Xu Ji, and Jiaoyang Huang. How does information bottleneck help deep learning? In *International Conference on Machine Learning (ICML)*, 2023.

[26] Vignesh Kothapalli and Tom Tirer. Kernel vs. kernel: Exploring how the data structure affects neural collapse. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2024.

[27] Vignesh Kothapalli, Tom Tirer, and Joan Bruna. A neural collapse perspective on feature evolution in graph neural networks. In *Conference on Neural Information Processing Systems (NeurIPS)*, volume 36, 2023.

[28] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

[29] Daniel Kunin, Atsushi Yamamura, Chao Ma, and Surya Ganguli. The asymmetric maximum margin bias of quasi-homogeneous neural networks. In *International Conference on Learning Representations (ICLR)*, 2022.

[30] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 1998.

[31] Xiao Li, Sheng Liu, Jinxin Zhou, Xinyu Lu, Carlos Fernandez-Granda, Zhihui Zhu, and Qing Qu. Principled and efficient transfer learning of deep models via neural collapse. In *Conference on Parsimony and Learning (Recent Spotlight Track)*, 2023.

[32] Zexi Li, Xinyi Shang, Rui He, Tao Lin, and Chao Wu. No fear of classifier biases: Neural collapse inspired federated learning with synthetic and fixed classifier. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[33] Yong Lin, Lu Tan, Yifan Hao, Honam Wong, Hanze Dong, Weizhong Zhang, Yujiu Yang, and Tong Zhang. Spurious feature diversification improves out-of-distribution generalization. In *International Conference on Learning Representations (ICLR)*, 2024.

[34] Haixia Liu. The exploration of neural collapse under imbalanced data. *arXiv preprint arXiv:2411.17278*, 2024.

[35] Xixi Liu, Yaroslava Lochman, and Christopher Zach. Gen: Pushing the limits of softmax-based out-of-distribution detection. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[36] Jianfeng Lu and Stefan Steinerberger. Neural collapse under cross-entropy loss. *Applied and Computational Harmonic Analysis*, 59, 2022.

[37] Jiawei Ma, Chong You, Sashank J Reddi, Sadeep Jayasumana, Himanshu Jain, Felix Yu, Shih-Fu Chang, and Sanjiv Kumar. Do we need neural collapse? Learning diverse features for fine-grained and long-tail classification. *openreview*, 2023.

[38] Wojciech Masarczyk, Mateusz Ostaszewski, Ehsan Imani, Razvan Pascanu, Piotr Miłoś, and Tomasz Trzcinski. The tunnel effect: Building data representations in deep neural networks. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2023.

[39] Dustin G Mixon, Hans Parshall, and Jianzong Pi. Neural collapse with unconstrained features. *Sampling Theory, Signal Processing, and Data Analysis*, 20(2):11, 2022.

[40] Vardan Papyan, X. Y. Han, and David L Donoho. Prevalence of neural collapse during the terminal phase of deep learning training. In *Proceedings of the National Academy of Sciences (PNAS)*, volume 117, 2020.

[41] Akshay Rangamani and Andrzej Banburski-Fahey. Neural collapse in deep homogeneous classifiers and the role of weight decay. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022.

[42] Akshay Rangamani, Marius Lindegaard, Tomer Galanti, and Tomaso A Poggio. Feature learning in deep classifiers through intermediate neural collapse. In *International Conference on Machine Learning (ICML)*, 2023.

[43] Mariia Seleznova, Dana Weitzner, Raja Giryes, Gitta Kutyniok, and Hung-Hsu Chou. Neural (tangent kernel) collapse. In *Conference on Neural Information Processing Systems (NeurIPS)*, volume 36, 2023.

[44] Jingtong Su, Ya Shi Zhang, Nikolaos Tsilivis, and Julia Kempe. On the robustness of neural collapse and the neural collapse of robustness. *Transactions on Machine Learning Research (TMLR)*, 2024.

[45] Peter Súkeník, Marco Mondelli, and Christoph H. Lampert. Deep neural collapse is provably optimal for the deep unconstrained features model. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2023.

[46] Peter Súkeník, Marco Mondelli, and Christoph H. Lampert. Neural collapse versus low-rank bias: Is deep neural collapse really optimal? *Conference on Neural Information Processing Systems (NeurIPS)*, 2024.

[47] Christos Thrampoulidis. Implicit optimization bias of next-token prediction in linear models. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2024.

[48] Christos Thrampoulidis, Ganesh Ramachandra Kini, Vala Vakilian, and Tina Behnia. Imbalance trouble: Revisiting neural-collapse geometry. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2022.

[49] Tom Tirer and Joan Bruna. Extended unconstrained features model for exploring deep neural collapse. In *International Conference on Machine Learning (ICML)*, 2022.

[50] Tom Tirer, Haoxiang Huang, and Jonathan Niles-Weed. Perturbation analysis of neural collapse. In *International Conference on Machine Learning (ICML)*, 2023.

[51] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[52] Haoqi Wang, Zhizhong Li, and Wayne Zhang. Get the best of both worlds: Improving accuracy and transferability by grassmann class representation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[53] Peng Wang, Huikang Liu, Can Yaras, Laura Balzano, and Qing Qu. Linear convergence analysis of neural collapse with unconstrained features. In *NeurIPS Workshop on Optimization for Machine Learning (OPT)*, 2022.

[54] Peng Wang, Xiao Li, Can Yaras, Zhihui Zhu, Laura Balzano, Wei Hu, and Qing Qu. Understanding deep representation learning via layerwise feature compression and discrimination. *arXiv preprint arXiv:2311.02960*, 2023.

[55] Sicong Wang, Kuo Gai, and Shihua Zhang. Progressive feedforward collapse of resnet training. *arXiv preprint arXiv:2405.00985*, 2024.

[56] Zijian Wang, Yadan Luo, Liang Zheng, Zi Huang, and Mahsa Baktashmotlagh. How far pre-trained models are from neural collapse on the target dataset informs their transferability. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[57] E Weinan and Stephan Wojtowytsch. On the emergence of simplex symmetry in the final and penultimate layers of neural network classifiers. In *Mathematical and Scientific Machine Learning*, 2022.

[58] Diyuan Wu and Marco Mondelli. Neural collapse beyond the unconstrainted features model: Landscape, dynamics, and generalization in the mean-field regime. *arXiv preprint arXiv:2501.19104*, 2025.

[59] Robert Wu and Vardan Papyan. Linguistic collapse: Neural collapse in (large) language models. *arXiv preprint arXiv:2405.17767*, 2024.

[60] Yingwen Wu, Ruiji Yu, Xinwen Cheng, Zhengbao He, and Xiaolin Huang. Pursuing feature separation based on neural collapse for out-of-distribution detection. *CoRR*, 2024.

[61] Mengjia Xu, Akshay Rangamani, Qianli Liao, Tomer Galanti, and Tomaso Poggio. Dynamics in deep classifiers trained with the square loss: Normalization, low rank, neural collapse, and generalization bounds. In *Research*, volume 6, 2023.

[62] Jiawei Zhang, Yufan Chen, Cheng Jin, Lei Zhu, and Yuantao Gu. Epa: Neural collapse inspired robust out-of-distribution detector. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6515–6519. IEEE, 2024.

[63] Yize Zhao, Tina Behnia, Vala Vakilian, and Christos Thrampoulidis. Implicit geometry of next-token prediction: From language sparsity patterns to model representations. In *First Conference on Language Modeling*, 2024.

[64] Zhisheng Zhong, Jiequan Cui, Yibo Yang, Xiaoyang Wu, Xiaojuan Qi, Xiangyu Zhang, and Jiaya Jia. Understanding imbalanced semantic segmentation through neural collapse. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[65] Jinxin Zhou, Xiao Li, Tianyu Ding, Chong You, Qing Qu, and Zhihui Zhu. On the optimization landscape of neural collapse under MSE loss: Global optimality with unconstrained features. In *International Conference on Machine Learning (ICML)*, 2022.

[66] Jinxin Zhou, Chong You, Xiao Li, Kangning Liu, Sheng Liu, Qing Qu, and Zhihui Zhu. Are all losses created equal: A neural collapse perspective. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2022.

[67] Jianggang Zhu, Zheng Wang, Jingjing Chen, Yi-Ping Phoebe Chen, and Yu-Gang Jiang. Balanced contrastive learning for long-tailed visual recognition. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[68] Zhihui Zhu, Tianyu Ding, Jinxin Zhou, Xiao Li, Chong You, Jeremias Sulam, and Qing Qu. A geometric analysis of neural collapse with unconstrained features. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2021.

Appendix A. Additional related work

Relevance of NC. The NC phenomenon raised significant interest in the machine learning community from both theoreticians and practitioners, due to its high relevance in both areas. Theoreticians use it to improve generalization understanding [11, 56, 61] both in-distribution and in transfer learning, OOD detection [14], imbalanced learning understanding [64], theory of feature learning [27, 37], robustness [44], as well as representation learning itself [2, 4, 10, 38, 54]. In practice, neural collapse has implications on transfer learning [5, 31, 52], OOD detection [35, 60, 62], compression [25], performance improvement [8, 52] and other aspects [32, 33, 67].

Unconstrained features model (UFM). First introduced in [9, 39], the UFM has been widely analyzed in the literature. The optimality of NC in the UFM has been proved for CE loss [29, 36, 57], MSE loss [65] and other losses [66]. A line of work [7, 9, 18, 48] has focused on the class-imbalanced setting, formulating a generalized NC geometry and proving its optimality. The loss landscape of the UFM was shown to be benign in [23, 65, 68], and the emergence of NC in the UFM through gradient descent training was proved in [15, 23, 39, 53]. Several extensions of the UFM to non-standard settings have been considered, including GNNs [27], large number of classes [24], unconstrained features regressed to the input data [50] and regression [1]. Recently, the UFM has been used to describe a form of NC in language modeling, where each context (sample) can be followed by multiple continuations, making the labels effectively stochastic [47, 63]. NC has been considered also for more layers following empirical observations [10, 16, 21, 42] and accordingly, UFM was generalized to multiple linear layers in [6, 13, 34], two non-linear layers in [49] and multiple non-linear layers in [12, 45, 46].

Appendix B. Extended notations

Here we provide an extended version of the notations, covering a wider range of architectures and denoting a wider class of objects, which will be necessary for our proofs and additional results.

We study two different data formats and architectures. For ResNets, the input data and one-hot labels are $X_0 \in \mathbb{R}^{d_0 \times N}$ and $Y \in \mathbb{R}^{K \times N}$, where d_0 is the input dimension, N the number of samples and K the number of classes. For transformers, the input data and one-hot labels are $X_0 \in \mathbb{R}^{N \times V \times C}$ and $Y \in \mathbb{R}^{N \times K \times C}$, where C is the context length (number of tokens in the prompt) and V the vocabulary size (number of distinct tokens). We take $C = 1$ when the third dimension of X_0, Y is not used. If we index a matrix with three abstract indices, the last one is implicitly equal to 1. We assume a class-balanced setting, i.e., $NC = Kn$, where n is the number of samples per class. Unless stated otherwise, we use x_{ki} to indicate the i-th sample of the k-th class. For transformers, a sample corresponds to the position of each individual token and, thus, x_{ki} corresponds to a token position labeled as class k, with samples ordered arbitrarily. For additional notation regarding vision transformers, see Appendix G.

ResNets and transformers. Let σ denote the ReLU function. Denote by $\mathrm{LN}(\cdot)$ the output of a normalization layer that first subtracts the mean of each column of the input from itself and then divides each column by its standard deviation (if the input is a vector, it returns the normalized vector; if the input is a matrix or tensor, it returns the matrix or tensor with centered and normalized columns of the inner-most dimension matrices). Define also $\mathrm{id}(\cdot)$ as the identity mapping.

Definition 6 *An L-block ResNet with LayerNorm and one linear layer per block (later referred to as L-RN1) is defined as*

$$f_\theta = \mathrm{lin}_L \circ \mathrm{LN} \circ (\mathrm{id} + \sigma \circ \mathrm{lin}_{L-1}) \circ \mathrm{LN} \circ (\mathrm{id} + \sigma \circ \mathrm{lin}_{L-2}) \circ \cdots \circ \mathrm{LN} \circ (\mathrm{id} + \sigma \circ \mathrm{lin}_1) \circ \mathrm{LN} \circ \mathrm{lin}_0, \quad (6)$$

where $\mathrm{lin}_l(x) = W_l x + b_l$ for all $l \in \{0, \dots, L\}$ and θ is the collection of all learnable parameters. We denote as $X_1 = \mathrm{LN}(W_0 X_0 + b_0)$, $X_{l+1} = \mathrm{LN}(X_l + \sigma(W_l X_l + b_l))$ ($l \in \{1, \dots, L-1\}$), $f_\theta(X_0) = X_{L+1} := W_L X_L$ the intermediate representations of the training data stored in a matrix form. We assume that all intermediate representations X_l ($l \in \{1, \dots, L\}$) are of dimension d. Analogously, L-RN2 denotes a ResNet with two linear layers per block defined as

$$f_\theta = \mathrm{lin}_L \circ \mathrm{LN} \circ (\mathrm{id} + \mathrm{lin}_{L-1,2} \circ \sigma \circ \mathrm{lin}_{L-1,1}) \circ \cdots \circ \mathrm{LN} \circ (\mathrm{id} + \mathrm{lin}_{1,2} \circ \sigma \circ \mathrm{lin}_{1,1}) \circ \mathrm{LN} \circ \mathrm{lin}_0, \quad (7)$$

with $X_1 = \mathrm{LN}(W_0 X_0 + b_0)$, $X_{l+1} = \mathrm{LN}(X_l + W_{l,2}\sigma(W_{l,1} X_l + b_{l,1}) + b_{l,2})$ ($l \in \{1, \dots, L-1\}$) and $f_\theta(X_0) = X_{L+1} := W_L X_L$.

Definition 7 *An L-block transformer with one or two linear layers in the attention sub-block and one or two layers in the MLP sub-block (later referred to as L-T11, L-T12, L-T21, L-T22 based on the number of linear layers in attention and MLP sub-blocks, respectively) is defined as*

$$f_\theta(Z) = \mathrm{lin}_{L+1} \circ \mathrm{LN}_{L+1} \circ \mathrm{B}_L \circ \cdots \circ \mathrm{B}_1 \circ \mathrm{Embed}(Z). \quad (8)$$

Here, $\mathrm{lin}_{L+1}(Z) = W_{L+1} Z + b_{L+1}$ is the last layer (b_{L+1} is a matrix with the same number of columns as Z that are all identical); $\mathrm{Embed}(Z) = W_e Z + W_p$ is the embedding layer with W_e being the token embedding and W_p (having the same shape as $W_e Z$) the positional embedding; and the l-th block is given by

$$\mathrm{B}_l = \mathrm{MLP}_l \circ \mathrm{LN}_{l,2} \circ \mathrm{ATTN}_l \circ \mathrm{LN}_{l,1}\,. \quad (9)$$

Such block consists of the normalization layers $\mathrm{LN}_{l,1}, \mathrm{LN}_{l,2}$, the MLP

$$\mathrm{MLP}_l(Z) = Z + \sigma(W_l Z + b_l), \ \textit{or} \ \mathrm{MLP}_l(Z) = Z + W_{l,2}\sigma(W_{l,1} Z + b_{l,1}) + b_{l,2}, \quad (10)$$

respectively for the architecture L-Tx1 and L-Tx2, and the single-head attention

$$\begin{aligned} \mathrm{ATTN}_l(Z) &= Z + W_{VO} Z A_l(Z), \ A_l(Z) = \mathrm{softmax}(M + Z^T W_{QK} Z/\sqrt{d}), \\ \textit{or} \ \mathrm{ATTN}_l(Z) &= Z + W_O W_V Z A_l(Z), \ A_l(Z) = \mathrm{softmax}(M + Z^T W_K^T W_Q Z/\sqrt{d}), \end{aligned} \quad (11)$$

respectively for the architecture L-T1x and L-T2x. The matrix M is the masking matrix whose entries are $-\infty$ on the lower triangle and 0 on the upper triangle and the diagonal.

Regardless of the model, let $h_\theta(\cdot)$ be the output of the corresponding architecture before the last layer, i.e., the feature on which neural collapse is defined. We denote by x_{ki}^l the i-th sample of the k-th class in the l-th layer.

Appendix C. GUFM-relevant results

We start with a lemma showing that nearly-optimal solutions of the GUFM problem above must necessarily be close to the global optima.

Lemma 8 *We have*

$$\limsup_{\epsilon \to 0} \ \mathrm{distmax}(\mathcal{M}_{\epsilon}^{GUFM} \backslash \mathcal{M}_{0}^{GUFM}, \mathcal{M}_{0}^{GUFM}) = 0. \tag{12}$$

The proof is deferred to Appendix F. Next, we focus on CE and MSE loss, showing that the optima of the corresponding GUFMs (denoted by UFM-CE and UFM-MSE) exhibit NC.

Lemma 9 *Assume that only the samples within the same class are in relation $\mathcal{R}$. Then, the global optima $\mathcal{M}_0^{UFM\text{-}CE}$ and $\mathcal{M}_0^{UFM\text{-}MSE}$ are all perfectly collapsed, i.e., for all $(W, X) \in \mathcal{M}_0^{UFM\text{-}CE}$, $\mathrm{NC1}(W, X) = \mathrm{NC2A}(W, X) = \mathrm{NC3}(W, X) = 0$ and for all $(W, X) \in \mathcal{M}_0^{UFM\text{-}MSE}$, $\mathrm{NC1}(W, X) = \mathrm{NC2B}(W, X) = \mathrm{NC3}(W, X) = 0$. Conversely, for any feasible pair (W, X) s.t. $\mathrm{NC1}(W, X) = \mathrm{NC2A}(W, X) = \mathrm{NC3}(W, X) = 0$, there exists a unique scalar c s.t. $(cW, X) \in \mathcal{M}_0^{UFM\text{-}CE}$; and for any feasible pair (W, X) s.t. $\mathrm{NC1}(W, X) = \mathrm{NC2B}(W, X) = \mathrm{NC3}(W, X) = 0$, there exists a unique scalar c s.t. $(cW, X) \in \mathcal{M}_0^{UFM\text{-}MSE}$.*

The proof is deferred to Appendix F. For the CE loss, it is based on an adaptation of the results in [68]. For the MSE loss, we compute the global optima by lower-bounding the loss, solving the problem for the lower-bound and showing that the loss and its lower-bound agree at these optima.

Appendix D. Additional discussions related to main results

Rate of convergence. While the results are stated asymptotically for simplicity, one can readily recover a convergence rate of the global optimum to NC from the argument. In particular, since the total regularization of the layers scales as L^{-1}, the global optima can only be suboptimal w.r.t. the GUFM objective with the same scaling. Then, assuming a differentiable loss (e.g., CE or MSE), the distance from the optima scales as the inverse of the power in the Taylor approximation of the loss at the global optima in the flattest direction, up to logarithmic factors that come from making a finer approximation. Now, for the CE loss, the leading term is quadratic: by using the chain rule, the slope of CE at the optimum is non-zero, and the sum of exponentials of dot-products between X, W is quadratic as we approach the ETF. Thus, the convergence in distance is $\tilde{O}(L^{-1/2})$, where $\tilde{O}$ omits logarithmic factors. For the MSE loss we compute by error analysis in the proof of Lemma 9 that the convergence rate is also $\tilde{O}(L^{-1/2})$.

Language modeling. When considering the transformer architecture, we require the labels to be unique given a specific context. While this is a realistic assumption in vision or language classification tasks (e.g., sentiment analysis, harmful content classification, spam detection), it does not apply to language pretraining, where a single context may have many different continuations. In fact, in the setting of non-unique continuations, neural collapse is *not* to be expected, and the optimal structure was discussed [47, 63] by using a form of UFM. We remark that Theorem 4 shows that the optimal solutions identified in these works are exhibited by transformers, as long as they satisfy the conditions in (4). This is the case, for instance, in some symmetric settings, see Proposition 2 in [63] with a slight modification in the underlying UFM (the authors consider weight decay instead of norm

constraints on the features), where the optimal limiting solution is indeed collapse. In non-symmetric cases, while NC is not expected to be optimal (as in the case with class imbalance [48]), transformers still represent the optimal zero-mean solution of the underlying UFM, whatever that is. This allows future work to focus on solving the application-relevant UFM in the corresponding setting and then use Theorem 4 to conclude that the solutions are globally optimal end-to-end.

Deep neural collapse. Although our theory focuses on last-layer geometry, the analysis sheds some light on the collapse in the earlier layers as well. In particular, one can readily obtain that any finite number of layers at the end of the network converges to neural collapse (with the exception of NC3 which has a different formulation for multi-layer collapse). Note that adding a residual connection (as in ResNets and transformers) resolves the inconsistency of deep UFMs pointed out in [46], where it is shown that the global optima of the deep UFM in the multi-class setting do *not* exhibit neural collapse. In fact, the optimal solution of a deep UFM with residual connections is obtained by simply copying the shallow UFM in the first layer and setting all remaining layers to 0. We also remark that, from the argument of Theorem 4, it follows that the global optima of the last $\tilde{L}$ layers of the network ($\tilde{L}$ being a constant independent of L) converge to the global optima of the corresponding deep GUFM with residual connections and depth $\tilde{L}$.

In contrast, understanding the emergence of neural collapse for a small, but constant fraction of the final layers of the network appears to require a different approach. Intuitively, if the network starts processing all samples at once from some layer onwards (which is expected to improve the loss w.r.t. our construction), then the collapse is progressive and occurs to some extent already in a constant fraction of the final layers, see also the discussion in [55].

Appendix E. Deep double-layer architectures are collapsed at the global optimum with vanishing regularization

Let us now consider ResNets with two linear layers per block and transformers with two linear layers per MLP sub-block (the number of matrices in the attention sub-block does not affect the result). Then, we show that neural collapse is globally optimal, provided that the regularization strength in all layers except the last one decreases with the depth L.

Theorem 10 *Let the architecture be L-RN2 or L-Tx2 for $x \in \{1, 2\}$. Assume the inner dimension of the L-Tx1 is at least $2V + 2$ and the inner dimension of L-RN1 is at least 4. Consider the optimization problem*

$$\min_{\theta} \mathcal{L}(f_\theta(X), Y) + \frac{\lambda_L}{2} \|W_L\|_F^2 + \frac{\lambda(L)}{2} \left\|\bar{\theta}\right\|^2, \tag{13}$$

where λ_L is a regularization on the weight matrix of the last layer that does not depend on L and $\lambda(L)$ is a depth-dependent regularization s.t. $\lambda(L) = o(\log(L)^{-1})$. Consider the corresponding GUFM with regularization λ_L. If $\mathcal{L}^*_{GUFM} > 0$, then

$$\limsup_{L \to \infty} \operatorname{distmax}(\tilde{\mathcal{M}}_{\mathrm{L}} \backslash \mathcal{M}_0^{GUFM}, \mathcal{M}_0^{GUFM}) = 0. \tag{14}$$

The reason why the regularization is required to be vanishing can be intuitively seen through a 1D example. Assume we want to represent a penultimate feature of size $\exp(a)$ with input 1 with a L-RN2 with inner dimension 1 and without LayerNorm. If x denotes a shared weight across all

residual layers, the output of the residual block will be $(1+x^2)^L$. In order to ensure that $(1+x^2)^L$ converges to $\exp(a)$ as $L \to \infty$, one needs to pick $x = \frac{\sqrt{a}}{\sqrt{L}}$, which implies that the sum of squares $\sum_{l=1}^{L} \left(\frac{\sqrt{a}}{\sqrt{L}}\right)^2$ is of constant order w.r.t. L. A similar example would also work with ResNets with LayerNorms in a setting which is at least bi-dimensional. In fact, the requirement on vanishing regularization is necessary for the statement to be true, and the result also cannot hold if both $\lambda(L)$ and λ_L are vanishing. An additional discussion on this point, together with a concrete dataset for which collapse cannot be reached, are provided in Appendix H. Understanding the structure of the optimal representations for double-layer architectures in the regime of constant regularization represents an exciting future direction.

The proof of Theorem 10 is similar to that of Theorem 4. In particular, the first layers of the blocks are defined in the same way, and the second layers are set to act as a projection matrix on the space spanned by the output of the first layer, which has rank 1. Furthermore, the scalings of these layers are split in identical square roots of the scaling of the original layer. Thus, the sum over the squared Frobenius norms is constant w.r.t. L, which requires $\lambda(L)$ to vanish in L. The detailed proof is deferred to Appendix F. We conclude this appendix by stating the approximate optimality of NC in the global optima of double-layer architectures under CE or MSE loss.

Corollary 11 *Let the architecture be L-RN2 or L-Tx2 for $x \in \{1, 2\}$. Assume the training data (X, Y) satisfies the assumptions of Corollary 5. Using CE or MSE loss, all global optima of the optimization problem (13) exhibit approximate neural collapse which gets tighter as L increases.*

Appendix F. Deferred proofs

Proof of Lemma 8. Assume by contradiction there exists a sequence $(X_n, W_n)_{n=1}^{\infty}$ of points such that

$$\lim_{n\to\infty} \mathcal{L}_{\text{GUFM}}(W_n, X_n) = \mathcal{L}^*_{\text{GUFM}},$$

but $\limsup_{n\to\infty} \operatorname{dist}((W_n, X_n), \mathcal{M}_0^{\text{GUFM}}) = c > 0$. Then, since the feasible set of GUFM is compact (for W, take a large-enough ball around 0 that must contain the global optimum), we can choose a subsequence $(X_{n_k}, W_{n_k})_{k=1}^{\infty}$ having an accumulation point $(\bar{W}, \bar{X})$ in the feasible set and s.t. $\operatorname{dist}((\bar{W}, \bar{X}), \mathcal{M}_0^{\text{GUFM}}) > 0$ (first picking a subsequence for which the limsup above is realized and only choosing a subsequence with accumulation point from this subsequence; then using the continuity of the distance to conclude). From the continuity of the loss function, it must follow $\mathcal{L}_{\text{GUFM}}(\bar{W}, \bar{X}) = \mathcal{L}^*_{\text{GUFM}}$, which also implies $(\bar{W}, \bar{X}) \in \mathcal{M}_0^{\text{GUFM}}$. However, this is a contradiction because the distance of this point from $\mathcal{M}_0^{\text{GUFM}}$ is both 0 and bigger than 0. ■

Proof of Lemma 9. For both losses, we will relax the problem and ignore the constraints coming from the equivalence relation $\mathcal{R}$. Then, we prove that NC1 holds in all of these cases, which grants equivalence between the relaxed and original problem.

For the CE loss, we apply Theorem 3.1 of [68]. In particular, from this theorem it follows that the optimal solutions of the regularized UFM-CE (not a-priori equivalent to (4) because of the feature constraint) exhibit neural collapse. From their proof, it is also clear that not only does the ratio between the sizes of the optimal w_k and x_{ki} only depend on the ratio of the regularization terms, but also that the absolute size of these vectors is an increasing function of the regularization strength, with the limit as $\lambda \to \infty$ being infinity. Therefore, let us pick λ_W from the paper to be λ in (4),

while we find λ_H s.t. the optimal solutions of the problem in [68] have norm $\sqrt{d}$. Then, the global optima of the regularized UFM-CE are exactly those of the UFM-CE we consider in (4).

To see the last statement, assume by contradiction that there is a global optimum of the problem in (4) which is not a global optimum of the regularized UFM-CE. Then, we can plug this solution into the regularized UFM-CE. Since it is not a global optimum, there exists a solution with strictly lower loss, and this optimum is guaranteed to have unit norm features. By plugging this optimum into (4), we must obtain a loss that is better than the optimal one, since the objectives are equivalent in this case. This leads to a contradiction. Similar arguments give that there cannot exist a global optimum of the regularized UFM-CE which is not a global optimum of (4), thus proving the desired equivalence.

For the MSE loss, we perform a direct computation which includes a perturbation analysis. To simplify the loss landscape, we start by defining a lower bound on the UFM-MSE loss, which we will analyze first. Denote

$$\underline{\mathcal{L}}_{\text{UFM-MSE}} := \frac{1}{2N} \sum_{k,i=1}^{K,n} \left(w_k^T x_{ki} - 1\right)^2 + \frac{\lambda}{2} \|W\|_F^2 \tag{15}$$

and $\underline{\mathcal{M}}_{\epsilon}^{\text{UFM-MSE}}$ the corresponding near-optimal set. Note that (15) is separable in the index k, thus we are facing K identical, independent optimization problems. We will now do a series of partial conditional optimizations and comment on the cost of deviating from these conditional optima. First, conditioning on any w_k (corresponding to the k-th row of W), we can almost exactly specify the optimal values of x_{ki} for any i. In particular, if $\|w_k\| \leq d^{-\frac{1}{2}}$, then the optimal solution is $x_{ki} = \sqrt{d} \cdot w_k / \|w_k\|$. If $\|w_k\| > d^{-\frac{1}{2}}$, then the optimal solution is any vector on a hypersphere such that $w_k^T x_{ki} = 1$. In the former case, for each x_{ki}, a deviation from the optimal value of the dot-product $w_k^T x_{ki} = \sqrt{d}\|w_k\|$ results in a quadratic increase in the loss around the optimal point (the cosine function has zero linear term in the Taylor expansion and non-zero quadratic term) or quartic if $\|w_k\| = d^{-\frac{1}{2}}$ (because the loss at optimum would be 0 and being itself a quadratic function, the effects would multiply). In the case $\|w_k\| > d^{-\frac{1}{2}}$, the loss increase around the optimum is again quadratic. Therefore, in all cases the maximum allowed deviation from the optimum given an extra loss of ϵ is at most $\mathcal{O}(\epsilon^{1/4})$ and, thus, goes to 0 as ϵ goes to zero.

Now, denote $z \equiv \|w_k\|$. The loss of the k-th group only depends on z and x_{ki}, but plugging-in the optimal value after solving for x_{ki} we arrive at a single-dimensional objective that only depends on z:

$$\frac{1}{2K}(1 - z\sqrt{d}) \max(1 - z\sqrt{d}, 0) + \frac{\lambda}{2} z^2.$$

From the form of this optimization problem, it is clear that the unique global optimum is reached on $(0, d^{-\frac{1}{2}})$. The solution is simply $\frac{1}{\sqrt{d}(1+\lambda K)}$. First, we note that, for fixed λ, K, this solution is strictly smaller than $d^{-\frac{1}{2}}$ with non-zero margin. Second, any deviation from this optimal solution will result in a quadratic increase in the loss function, therefore for a fixed extra loss of ϵ, the maximum allowed deviation of $\|z_k\|$ from its optimal value is $\mathcal{O}(\epsilon^{1/2})$, which also goes to 0 with ϵ going to 0. Moreover, since its optimal value (and also maximum allowed deviation for ϵ small-enough) is strictly smaller than $d^{-\frac{1}{2}}$, we know that the optimal value of the x_{ki} is indeed $\sqrt{d} \cdot w_k / \|w_k\|$ and the maximum allowed deviation is also $\mathcal{O}(\epsilon^{1/2})$.

The function value in (15) cannot be optimized any further, thus we know what $\underline{\mathcal{M}}_0^{\text{UFM-MSE}}$ is. In particular, the solutions in $\underline{\mathcal{M}}_0^{\text{UFM-MSE}}$ must satisfy the NC1 and NC3 properties. Now, if the global

optima of (4) with MSE loss and (15) are equal, then $\mathcal{M}_0^{\text{UFM-MSE}} \subset \underline{\mathcal{M}}_0^{\text{UFM-MSE}}$ and thus the optimal solutions of (4) with MSE loss must also satisfy the NC1 and NC3 criteria from the lemma statement.

To show that the global optima are equal and to argue about NC2, we turn back to the original problem (4) with MSE loss. Since we know that the optimal solutions agree, we can focus directly on $\underline{\mathcal{M}}_0^{\text{UFM-MSE}}$. After plugging any optimal solution of (15) into $\mathcal{L}_{\text{UFM-MSE}}$, we see that the regularization part is constant, so we are left with optimizing the fit part. Analyzing the loss incurred by x_{ki} on position $l \neq k$ we see that it is $(w_l^T x_{ki})^2 = (w_l^T w_k)^2 d(1+\lambda K)^2$. Summing this over all indices and samples (using the symmetries) we see that the total loss is proportional to the Frobenius norm of the off-diagonal elements of WW^T. Therefore, a lower-bound on the loss is 0, which is achievable provided W has at least as many columns as rows, as assumed in the lemma. Let us simply choose W to be a scaled orthogonal matrix, and note that the loss cannot be optimized any further. Thus, we see that $\mathcal{L}^*_{\text{UFM-MSE}} = \underline{\mathcal{L}}^*_{\text{UFM-MSE}}$ and the solutions of (4) with MSE must satisfy NC2. Any deviation of W from an orthogonal matrix will result in an increase in the loss which is at least quartic: given a fixed extra loss of ϵ, the solution in $\mathcal{M}_\epsilon^{\text{UFM-MSE}}$ must be $\mathcal{O}(\epsilon^{1/4})$ close to an orthogonal matrix.

Finally, the converse statements also readily follow from the above computations. ■

Proof of Theorem 4. We first discuss how to deal with the equivalence relation $\mathcal{R}$. The argument is identical whether we take individual samples if all samples are distinct, or we treat the equivalence classes as individual samples. Thus, for simplicity of notation we assume, without loss of generality, that the samples are all distinct.

We start with the proof for the L-RN1 model. Notice that $\mathcal{L}_{L,1}(\theta) = \mathcal{L}_{\text{GUFM}}(W_L, X_L) + \frac{\lambda}{2}\sum_{l=1}^{L-1}\|W_l\|_F^2$. The goal is to show $\mathcal{L}^*_{\text{GUFM}} = \lim_{L\to\infty} \mathcal{L}^*_{L,1}$. In that case, $\mathcal{L}_{\text{GUFM}}(W_L, X_L)$ must converge to $\mathcal{L}^*_{\text{GUFM}}$. Therefore, (W_L, X_L) induced by $\theta \in \mathcal{M}_0^{L,1}$ must also belong to $\mathcal{M}_\epsilon^{\text{GUFM}}$ for ϵ arbitrarily small, which evoking Lemma 8 guarantees the convergence as defined in (5).

Note that we can represent a one-block-deeper ResNet that perfectly copies the original ResNet by simply adding an identity block with zero weight matrices/biases and residual connection left untouched. Thus, $\mathcal{L}^*_{L,1}$ is non-increasing in L and it suffices to prove the limit for any sequence of L's going to infinity. We will prove it by explicitly constructing a sequence of L-RN1 ResNets s.t. their losses converge to $\mathcal{L}^*_{\text{GUFM}}$ as $L \to \infty$.

Pick any $(W_L, X_L) \in \mathcal{M}_0^{\text{GUFM}}$ and relabel $H := X_L$. Thus, h_{ki} is the feature representation of the sample ki in the penultimate layer. Define $\bar{H}$ as the matrix of unique points h_{ki}, and let us index them with a single index as $\bar{h}_j$. Denote the number of these unique points as $\bar{K}$. If we write $j(ki)$ we mean the index j such that $\bar{h}_j = h_{ki}$. Before starting the construction, we need to define a key data-dependent quantity. First, take $X_1 = \text{LN}(W_0 X_0 + b_0)$ for b_0 and W_0 sampled from a continuous distribution. Since points in X_0 are all disjoint, this property holds also for X_1 with probability 1. Moreover, with probability zero, any sample in X_1 is identical to h_{ki} for any of the vectors in H. For simplicity, we will refer to X_1 and its samples as X and drop the index. Fix an ordering of the points x_{ki} as the lexicographical ordering of (k, i). For each (k, i) find a smooth oriented curve $\mathcal{G}_{ki}$ connecting x_{ki} with h_{ki} on the set of feasible points ($\sqrt{d}$ norm hypersphere with zero-sum entries) such that all of the following holds:

1. The curvature of $\mathcal{G}_{ki}$ defined as the Lipschitz constant of the unit-norm oriented tangent function $\mathcal{T}_{ki}$ is bounded by B.
2. For all $(l, j) > (k, i)$, $\max_{x\in\mathcal{G}_{ki}} x_{lj}^T x \leq d(1-m)$ for some $m > 0$, i.e., all the subsequent points x_{lj} are far enough from the curve $\mathcal{G}_{ki}$.

3. There is precisely one point $\bar{x}_{ki} \in \mathcal{G}_{ki}$ such that $\bar{x}_{ki}^T h_{ki} = d(1-cm)$, where $c > 1$ is chosen large enough. Denote $\bar{\mathcal{G}}_{ki}$ as the set of points on $\mathcal{G}_{ki}$ between x_{ki} and $\bar{x}_{ki}$. Then, we assume that, for all $(l,j) < (k,i)$, $\max_{x \in \bar{\mathcal{G}}_{k,i}, y \in \mathcal{G}_{l,j} \setminus \bar{\mathcal{G}}_{l,j}} x^T y \leq d(1-m)$.

4. The length of $\mathcal{G}_{ki}$ is no more than $2\pi\sqrt{d}$.

5. m is chosen small enough s.t. $10cm \leq (d - \max_{j(ki) \neq j(lp)} \bar{h}_{j(ki)}^T \bar{h}_{j(lp)})/d$.

It is clear that a construction satisfying these properties exists, since the constants B, c, m are chosen with respect to X, H and the number of points we consider is finite. We also note that this requires the inner dimension of the representations to be at least 4 since this would not be possible on a 2D circle.

The idea of the construction is as follows. Take L large enough and divide the layers into $N + \bar{K} + 1$ blocks. The first N blocks are of the same number of layers L_1, and the depth of the last one will be specified later. Each of the first N blocks of layers will focus on a single sample, while not changing the representation of the other samples at all. The goal of the ki-th block is to only move the ki-th sample on its curve towards h_{ki}, until it hits $\bar{x}_{ki}$. Then, the $\bar{K}$ next blocks of depth L_2 will move all the samples corresponding to the same $j(ki)$ at once, ever closer to their respective $\bar{h}_{j(ki)}$ vectors. Finally, the very last block which consists of the very last layer will simply be chosen as the optimal W_L corresponding to H.

We will now construct explicitly all the layers. Denote by W_{ki}^l, b_{ki}^l the parameters of the l-th layer of the ki-th block and define x_{ki}^l to be the feature representation of the ki-th sample as an input to that layer. Consider a sphere with center x_{ki}^l and radius $\frac{\alpha_{ki}^l m\sqrt{d}}{2\sqrt{d+m^2/4}}$, where α_{ki}^l is a small-enough number whose role will be clear soon. Since this sphere is small enough and $\mathcal{G}_{ki}$ has bounded curvature, there exists exactly one point $\tilde{x}_{ki}^{l+1}$ on the intersection between $\mathcal{G}_{ki}$ and the considered sphere which is closer to h_{ki} as x_{ki}^l. Denote $d_{ki}^l = \frac{\tilde{x}_{ki}^{l+1} - x_{ki}^l}{\|\tilde{x}_{ki}^{l+1} - x_{ki}^l\|} = \frac{\tilde{x}_{ki}^{l+1} - x_{ki}^l}{\frac{\alpha_{ki}^l m\sqrt{d}}{2\sqrt{d+m^2/4}}}$. The weights are constructed as follows:

$$W_{ki}^l = \alpha_{ki}^l \frac{\mathbf{1} + \frac{m}{2} d_{ki}^l}{\sqrt{d+m^2/4}} \frac{(x_{ki}^l)^T}{\sqrt{d}}, \tag{16}$$
$$b_{ki}^l = -\left(1 - \frac{m}{2}\right) \frac{\alpha_{ki}^l \sqrt{d}}{\sqrt{d+m^2/4}} \mathbf{1},$$

if $(x_{ki}^l)^T h_{ki} \leq d(1-cm)$, otherwise $W_{ki}^l = 0; b_{ki}^l = 0$. The α_{ki}^l is an optimizable parameter and since the form above is also W's SVD, it is its singular value. Thus, $\sigma(W_{ki}^l x_{ki}^l + b_{ki}^l) = \frac{\alpha_{ki}^l m\sqrt{d}}{2\sqrt{d+m^2/4}}(\mathbf{1} + d_{ki}^l)$, while $\sigma(W_{ki}^l x_{st}^l + b_{ki}^l) = 0$ for any $(s,t) \neq (k,i)$ thanks to our margin definition. Therefore, before x_{ki} hits its final destination, we have

$$x_{ki}^{l+1} = \mathrm{LN}\left(x_{ki}^l + \frac{\alpha_{ki}^l m\sqrt{d}}{2\sqrt{d+m^2/4}}(\mathbf{1} + d_{ki}^l)\right) = \frac{\sqrt{d}\left(x_{ki}^l + \frac{\alpha_{ki}^l m\sqrt{d}}{2\sqrt{d+m^2/4}} d_{ki}^l\right)}{\left\|x_{ki}^l + \frac{\alpha_{ki}^l m\sqrt{d}}{2\sqrt{d+m^2/4}} d_{ki}^l\right\|} = \tilde{x}_{ki}^{l+1}.$$

From this, it is clear that x_{ki} is moving along and on the curve, while the other samples stay stationary.

It remains to compute how fast x_{ki} travels along the geodesic with this construction. To this end, denote $\beta_{ki}^l := \sphericalangle(x_{ki}^l, \bar{x}_{ki})$ as the spherical angle between x_{ki}^l and $\bar{x}_{ki}$. Let $\Delta\beta_{ik}^l := \beta_{ki}^{l+1} - \beta_{ki}^l$, i.e., the angle shift of x_{ki}^l in the l-th layer of the ki-th block. Using simple trigonometry we can compute:

$$\Delta\beta_{ik}^l = 2\arcsin\left(\frac{\alpha_{ki}^l m}{2\sqrt{d+m^2/4}}\right) \geq \frac{m}{4\sqrt{d}}\alpha_{ki}^l,$$

where the inequality holds for α_{ki}^l small enough.

Therefore, it suffices to choose L and L_1 large enough and set $\alpha_{ki}^l = \frac{4\sqrt{d}\beta_{ki}^l}{L_1 m}$ if $(x_{ki}^l)^T h_{ki} \leq d(1-cm)$ and 0 otherwise. In this way, the total regularization cost of the layers in the first N blocks can be upper bounded as

$$\frac{\lambda}{2}\sum_{k,i,l}^{K,n,L_1}\left\|W_{ki}^l\right\|_F^2 \leq \frac{32d\pi^2\lambda N}{L_1 m^2}.$$

We see that this cost goes to 0 as L_1 goes to infinity.

After N blocks, all the samples now lie within the c-multiple of margin ($(x_{ki}^{L_1})^T h_{ki} \geq d(1-cm)$) of their respective optimal h_{ki} features. The goal of each of the $\bar{K}$ blocks is to move the corresponding samples in the j-th group all together ever closer to these final vectors. Since this time the construction will be equivalent for all the samples within one group, we will refer to these samples simply as a single j-th sample in the l-th layer of the respective block, using the notation x_j^l. We define all layers in the j-th block as follows:

$$W_j^l = \alpha_j^l \frac{\mathbf{1} + cm\bar{h}_j}{\left\|\mathbf{1} + cm\bar{h}_j\right\|}\frac{\bar{h}_j^T}{\sqrt{d}},$$
$$b_j^l = -(1-2cm)\frac{\alpha_j^l\sqrt{d}}{\left\|\mathbf{1} + cm\bar{h}_j\right\|}\mathbf{1},$$

where again α_j^l is an optimizable parameter. By similar computations as above, the above construction makes sure that $\sigma(W_j^l x_j^l + b_j^l) = \frac{\alpha_j^l}{\left\|\mathbf{1}+cm\bar{h}_j\right\|}((\bar{h}_j^T x_j^l - d + 2cmd)\mathbf{1} + cm\bar{h}_j^T x_j^l \bar{h}_j)$ while $\sigma(W_j^l x_i^l + b_j^l) = 0$ for $i \neq j$. After subtracting the mean in the layer norm we are adding $\frac{\alpha_j^l cm\bar{h}_j^T x_j^l}{\left\|\mathbf{1}+m\bar{h}_j\right\|}\bar{h}_j$, which is at least a $\frac{\alpha_j^l cm}{2\sqrt{d}}$ multiple of $\bar{h}_j$. Denote $\beta_j^l = \sphericalangle(x_j^l, \bar{h}_j)$ and $\Delta\beta_j^l = \beta_j^{l+1} - \beta_j^l$.

Using trigonometry again, we get:

$$\Delta\beta_j^l \geq \arctan\left(\frac{\alpha_j^l cm\sin(\beta_j^l)}{2\sqrt{d}(1+\alpha_j^l cm\cos(\beta_j^l)/(2\sqrt{d}))}\right) \geq \frac{\alpha_j^l cm\sin(\beta_j^l)}{4\sqrt{d}(1+\alpha_j^l cm\cos(\beta_j^l)/(2\sqrt{d}))} \geq \frac{\alpha_j^l cm\beta_j^l}{16\sqrt{d}},$$

where all inequalities hold from basic properties of trigonometric functions for small-enough angles. Thus, the angular shift is lower bounded as follows: $\Delta\beta_r^l \geq \frac{\alpha_j^l cm\beta_j^l}{16\sqrt{d}}$. If we choose $\alpha_j = \alpha_j^l$ constant across layers, we get $\beta_j^{L_2} \leq \left(1 - \frac{\alpha_j cm}{16\sqrt{d}}\right)^{L_2}\beta_j^0$.

We will choose $\alpha_j = \frac{16\sqrt{d}\log(L_2)}{cmL_2}$. Then, the total regularization of the layers in the penultimate blocks is upper bounded as follows:

$$\frac{\lambda}{2}\sum_{l,j=1}^{L_2,\bar{K}} \left\|W_j^l\right\|_F^2 \leq \frac{2^7 Nd\lambda \log(L_2)^2}{m^2 L_2}.$$

This goes to zero linearly up to poly-log factors as L_2 goes to infinity. Finally, we have that the final positions $x_{ki}^{L_2}$ of the samples converge fast to their optimal counterparts h_{ki} with L. To see this, plugging our choice of α_j into $\beta_j^{L_2} \leq \left(1 - \frac{\alpha_j cm}{16\sqrt{d}}\right)^{L_2} \beta_j^0$ we get $\beta_j^{L_2} \leq \left(1 - \frac{\log(L_2)}{L_2}\right)^{L_2} \beta_j^0 \leq \frac{2\beta_r^0}{L_2}$, so the samples converge linearly to their optimal positions as $L_1, L_2 \to \infty$. From the continuity of the fit part of the loss, we see that the total loss of this construction indeed converges to $\mathcal{L}^*_{\text{GUFM}}$ of the corresponding GUFM problem. Therefore, our upper bound on the loss of globally optimal solutions converges to $\mathcal{L}^*_{\text{GUFM}}$ and evoking Lemma 8 we know that a (W_L, X_L) optimal for (3) is nearly optimal for (4) and thus exhibits the required convergence.

Next, we continue with the proof for L-Tx1. Notice that, if we can ensure after the end of the first block that all the *different* contexts have different representations and that two representations of different contexts don't lie on a line with some of the final positions h_{ki}, then by setting all the weights in attention layers of the subsequent blocks to 0, the rest of the transformer becomes a ResNet with LayerNorm and we can apply an identical construction as in the L-RN1 part to conclude. The only caveat (except making sure that the margin is positive) is that, since the total regularization loss of the construction for L-RN1 goes to 0 with L going to infinity, we must make sure that the same is true for the first block. However, as we will see, this will make the margin m a function of L that slowly goes to 0. To compensate for this, we will need to set the layers in the subsequent blocks accordingly bigger, and we will make sure that the margin goes to 0 slowly enough so that this adjustment will not qualitatively change the results. Another issue we have to deal with is that if the norm of the W_{QK} matrix has to go to 0, the attention weights must necessarily converge to uniform. Thus, our construction must withstand this burden.

We will start with the construction of the embedding matrices. The embedding matrix $W_e \in \mathbb{R}^{d\times d_0}$ will just lift the dimension to the inner-dimension of the transformer $d_l \geq 2V + 2$, i.e., the v-th column of W_e is e_v in d_l-dimensional space. Then, the $(C-i)$-th column of $W_p \in \mathbb{R}^{d\times C}$ will be $a \cdot e_{2V+1} + b \cdot e_{2V+2}$, where $a, b > 0$ are the unique solutions of the following two equations: $a + b = -1$ and $a^2 + b^2 = 2^{2(i+1)} - 1$. Thus, after the embedding layer, the sum of the entries of the entire embedding is 0 and after the first normalization layer, the j-th token at the $(C-i)$-th position will have $\sqrt{d}2^{-(i+1)}$ on its j-th entry and the only other non-zero entries will be at positions $2V+1$ and $2V+2$.

Let us construct the first block. Here, all MLP layers will be set to 0 so that they have zero effect. Moreover, due to the constraints discussed above, attention matrices W_K, W_Q or W_{QK} will also be set to zero. Finally, the value and output matrices will be set as $W_V = W_O = \sqrt{\gamma(L)}A$ or $W_{VO} = \gamma(L)A$, where A shifts all entries from the range $1, \ldots, V$ to the range $V+1, \ldots, 2V$, and $\gamma(L)$ is a decreasing function converging to 0 at infinity that will be defined later. This ensures that the representations before and after attention are summed in the residual connection on different positions, which will be technically convenient later. Since the attention matrices are identically zero, the attention weights corresponding to the c-th token will just be uniform $1/c$ for all the tokens up to this one. Therefore, the representation of the c-th token after the attention layer and before the

residual connection is the $\gamma(L)$-multiple of the average of all the representations of the previous tokens and itself from an input to the first block shifted by V positions.

We now show that two different contexts must necessarily have different representations, which gives that the margin after block 1 is non-zero. If we compare two samples (contexts) with different context lengths, then they will necessarily have different numbers of distinguishable summands (i.e. various negative powers of 2, divided by the sample's context length) present in the entries between $(V+1)$-th and $2V$-th. Since there is a different number of summands, there must exist at least one entry where the number of summands disagree, and the numbers in this entry must have different numbers of ones in their binary representation, which guarantees that samples with different context lengths must have different representations. Furthermore, two samples with the same context length but different contexts will be divided by the same averaging number, but then they can be distinguished since the map from contexts to representations (without dividing by the context length) is injective due to the uniqueness of the binary representation of the summands.

Therefore, all non-identical contexts have different representations and, in addition, the previous argument also shows that every pair of representations of two different contexts is linearly independent. This remains true after the residual connection. If we choose $\gamma(L)$ small enough for all L, then the original encodings of the current token will not mix up with the much smaller summands from the attention layer. The relative size of all the summands stays the same also after normalization and the MLP block has no effect, so all different contexts have different representations after the first layer. The only issue we could face is that the representations end up coinciding with one of the h_{ki}'s. To avoid this, W_O or W_{VO} can implement a tiny rotation. Since the number of tiny rotations is uncountably infinite, there is at least one for which there is no intersection. Let $\sqrt{d}\tilde{m}$ be the minimal distance between representations of any two samples after the attention mixing, before the multiplication by value and output matrices and before the residual connection. Note that $\tilde{m}$ is positive and independent of X, Y, L, because the different contexts are all pairwise linearly independent. Then, after the multiplication by the value and output matrices, such distance will be $\gamma(L)\sqrt{d}\tilde{m}$. For small enough $\gamma(L)$, the worst-case addition in the residual connection corresponds to the case in which the two samples with the same latest token also realize the margin minimum. However, if $\gamma(L) \leq 0.1$, then the difference of the samples after the residual connection and after the normalization is at least equal to the distance between the representations on positions $V+1$ to $2V$, which is at least $\gamma(L)\sqrt{d}\tilde{m}$. Thus, this is the minimum pairwise distance of the data after the first attention block.

Next, we can apply the construction for L-RN1 if we set all the remaining attention layers to zeros, since then the remainder of the network will be functionally equivalent to L-RN1. The only remaining issue is that the margin after the first layer is a function $\gamma(L)$ of the total number of layers. To choose a good scaling of $\gamma(L)$, we need to consider the elements of the construction for L-RN1 that depend on the margin, which is the sum of the Frobenius norms of the layers in the first $N + \bar{K}$ blocks. This is upper-bounded by $\frac{32\pi^2\lambda N d}{L_1 m^2} + \frac{128 N d \lambda \log(L_2)^2}{m^2 L_2}$. Therefore, if we choose $\gamma(L) = \Theta(L_1^{1/4}) = \Theta(L_2^{1/4})$, then both the sum of Frobenius norms of the layers in first N layer blocks, as well as the Frobenius norms of W_V, W_O or W_{VO} in the first block of the transformer will go to 0 as $L_1, L_2 \to \infty$. This concludes the proof. ■

Proof of Corollary 5. This is a straightforward combination of Lemma 9 and Theorem 4 once we use that identical contexts for transformers are only labeled by one class, which allows to directly apply the lemma. ■

Proof of Theorem 10. The proof follows that of Theorem 4. The only difference is that the construction of the weight matrices changes so that $W_{ki}^{l,1}$ and $b_{ki}^{l,1}$ have $\sqrt{\alpha_{ki}^{l}}$ in place of α_{ki}^{l}. The second layers' weight matrices $W_{ki}^{l,2}$ are defined as $\sqrt{\alpha_{ki}^{l}}$-multiples of the projection matrix on the span of the output of the first sub-layer on sample x_{ki}^{l}, so that the total mapping will be identical to the single-layer construction. Using analogous computations as above, we get:

$$\frac{\lambda(L)}{2}\sum_{k,i,l}^{K,n,L_1}\left\|W_{ki}^{l,1}\right\|_F^2+\left\|W_{ki}^{l,2}\right\|_F^2\leq\frac{16\sqrt{d}\pi\lambda(L)N}{m},$$

and for the second part of the blocks we get:

$$\frac{\lambda}{2}\sum_{l,j=1}^{L_2,\bar{K}}\left\|W_{j}^{l,1}\right\|_F^2+\left\|W_{j}^{l,2}\right\|_F^2\leq\frac{16N\sqrt{d}\log(L_2)\lambda(L)}{m}.$$

In order for the sum of these two components to go to zero, we need $\lambda(L)=o(\log(L_2)^{-1})$ and we can choose $L_1=\Theta(L_2)$. The rest of the proof is identical to that of Theorem 4. ■

Proof of Corollary 11. This is a straightforward combination of Lemma 9 and Theorem 10 once we use that identical contexts for transformers are only labeled by one class, which allows to directly apply the lemma. ■

Appendix G. Alternative architectures

G.1. Vision transformers

For vision transformers, the data is tensor-like $X_0\in\mathbb{R}^{N\times d_0\times C}$, where C now denotes the number of patches and d_0 is the dimension of the patch. However, the labels remain two-dimensional $Y\in\mathbb{R}^{N\times K}$. What is considered as a sample depends on how labels are produced in the transformer. The simplest option (w.r.t. the rest of our paper) is to generate the prediction on the last patch of the sequence, keeping the causal mask. This will, however, change the definition of "samples" and the NC metrics, since we only need to focus on the last patch. Therefore, samples will only be considered as the last patch, and the NC metrics will only be defined over the representations of the last patches. Similarly, the equivalent DUFM will also correspond to the last patches.

Theorem 4 and 10 and, thus, also Corollary 5 and 11 hold for vision transformers too, as long as we do the following changes to the proof of Theorem 4 (the other statements are adjustable trivially once this is established).

Necessary adjustments to the proof. Together with the uniqueness of the labeling function, we will also assume that the samples are taken from a continuous distribution (which is reasonable in the vision domain). This guarantees that the feature representations of the final patches are unique also after the first transformer block, as the event that averages over patches of two different samples coincide has zero probability. The rest of the proof is similar to that of Theorem 4, but the subsequent MLP layers only focus on the movement of the last patches' representations and the movement of the other patches is irrelevant.

G.2. Pre-LN ResNets and transformers

Unlike the post-LN ResNets (Definition 6) and transformers (Definition 7), the pre-LN architectures apply the LayerNorm directly before the attention and/or linear layers, but only *within* the residual connection, leaving the main residual stream untouched. While this potentially makes the features at initialization grow linearly with depth, it makes for more stable gradients thanks to the direct residual path, avoiding LayerNorms that can serve as error propagation channels. This significantly simplifies the training dynamics and therefore the pre-LN transformers are currently being predominantly used. For this reason, we fully define the pre-LN architectures here and then discuss in sufficient amount of detail how to adjust the proof for this setting, since the results are qualitatively the same.

Definition 12 *An L-block pre-LN ResNet with LayerNorm and one linear layer per block (later referred to as pre-L-RN1) is defined as*

$$f_\theta = \mathrm{lin}_L \circ \mathrm{LN} \circ (\mathrm{id} + \sigma \circ \mathrm{lin}_{L-1} \circ \mathrm{LN}) \circ (\mathrm{id} + \sigma \circ \mathrm{lin}_{L-2} \circ \mathrm{LN}) \circ \cdots \circ (\mathrm{id} + \sigma \circ \mathrm{lin}_1 \circ \mathrm{LN}) \circ \mathrm{LN} \circ \mathrm{lin}_0, \tag{17}$$

where $\mathrm{lin}_l(x) = W_l x + b_l$ for all $l \in \{0, \dots, L\}$ and θ is the collection of all learnable parameters. We denote as $X_1 = \mathrm{LN}(W_0 X_0 + b_0)$, $X_{l+1} = X_l + \sigma(W_l \,\mathrm{LN}(X_l) + b_l)$ ($l \in \{1, \dots, L-1\}$), $f_\theta(X_0) = X_{L+1} := W_L \,\mathrm{LN}(X_L)$ the intermediate representations of the training data stored in a matrix form. We assume that all intermediate representations X_l ($l \in \{1, \dots, L\}$) are of dimension d. Analogously, L-RN2 denotes a ResNet with two linear layers per block defined as

$$f_\theta = \mathrm{lin}_L \circ \mathrm{LN} \circ (\mathrm{id} + \mathrm{lin}_{L-1,2} \circ \sigma \circ \mathrm{lin}_{L-1,1} \circ \mathrm{LN}) \circ \cdots \circ (\mathrm{id} + \mathrm{lin}_{1,2} \circ \sigma \circ \mathrm{lin}_{1,1} \circ \mathrm{LN}) \circ \mathrm{LN} \circ \mathrm{lin}_0, \tag{18}$$

with $X_1 = \mathrm{LN}(W_0 X_0 + b_0)$, $X_{l+1} = X_l + W_{l,2}\sigma(W_{l,1} \,\mathrm{LN}(X_l) + b_{l,1}) + b_{l,2}$ ($l \in \{1, \dots, L-1\}$) and $f_\theta(X_0) = X_{L+1} := W_L \,\mathrm{LN}(X_L)$.

Definition 13 *An L-block pre-LN transformer with one or two linear layers in the attention sub-block and one or two layers in the MLP sub-block (later referred to as pre-L-T11, pre-L-T12, pre-L-T21, pre-L-T22 based on the number of linear layers in attention and MLP sub-blocks, respectively) is defined as*

$$f_\theta(Z) = \mathrm{lin}_{L+1} \circ \mathrm{LN}_{L+1} \circ \mathrm{B}_L \circ \cdots \circ \mathrm{B}_1 \circ \mathrm{LN}_0 \circ \mathrm{Embed}(Z). \tag{19}$$

Here, $\mathrm{lin}_{L+1}(Z) = W_{L+1} Z + b_{L+1}$ is the last layer (b_{L+1} is a matrix with the same number of columns as Z that are all identical); $\mathrm{Embed}(Z) = W_e Z + W_p$ is the embedding layer with W_e being the token embedding and W_p (having the same shape as $W_e Z$) the positional embedding; and the l-th block is given by

$$\mathrm{B}_l = (\mathrm{id} + \mathrm{MLP}_l \circ \mathrm{LN}_{l,2}) \circ (\mathrm{id} + \mathrm{ATTN}_l \circ \mathrm{LN}_{l,1}). \tag{20}$$

Such block consists of the normalization layers $\mathrm{LN}_{l,1}, \mathrm{LN}_{l,2}$, the MLP

$$\mathrm{MLP}_l(Z) = \sigma(W_l Z + b_l), \text{ or } \mathrm{MLP}_l(Z) = W_{l,2}\sigma(W_{l,1} Z + b_{l,1}) + b_{l,2}, \tag{21}$$

respectively for the architecture pre-L-Tx1 and pre-L-Tx2, and the single-head attention

$$\begin{aligned} \mathrm{ATTN}_l(Z) &= W_{VO} Z A_l(Z), \;\; A_l(Z) = \mathrm{softmax}(M + Z^T W_{QK} Z/\sqrt{d}), \\ \text{or } \mathrm{ATTN}_l(Z) &= W_O W_V Z A_l(Z), \;\; A_l(Z) = \mathrm{softmax}(M + Z^T W_K^T W_Q Z/\sqrt{d}), \end{aligned} \tag{22}$$

respectively for the architecture pre-L-T1x and pre-L-T2x. The matrix M is the masking matrix whose entries are $-\infty$ on the lower triangle and 0 on the upper triangle and the diagonal.

We note that the first LayerNorm right after embedding layer, which might not be used in practice often, is introduced for technical convenience but does not change the results qualitatively. Theorem 4 and 10 and, thus, also Corollary 5 and 11 hold for pre-LN architectures too, as long as we do the following changes to the proof of Theorem 4 (the other statements are adjustable trivially once this is established).

Necessary adaptations to the proof. This architecture has the disadvantage that it does not immediately absorb deviations from the zero-sum sphere and therefore, technically, the single linear layer architectures can only add non-negative changes to the residual stream. However, we argue that an *almost identical* construction to the one in proof of Theorem 4 works here as well. Note that the construction from this proof, see (16):

$$W_{ki}^l = \alpha_{ki}^l \frac{\mathbf{1} + \frac{m}{2} d_{ki}^l}{\sqrt{d + m^2/4}} \frac{(x_{ki}^l)^T}{\sqrt{d}},$$
$$b_{ki}^l = -\left(1 - \frac{m}{2}\right) \frac{\alpha_{ki}^l \sqrt{d}}{\sqrt{d + m^2/4}} \mathbf{1},$$

will result in a shift in x_{ki} that can be written as $\frac{\alpha_{ki}^l m\sqrt{d}}{2\sqrt{d+m^2/4}}(\mathbf{1} + d_{ki}^l)$ and the $\mathbf{1}$ will not get absorbed in the residual stream, but *is orthogonal* to the zero-sum component of the movement of x_{ki} *and* it *will* get absorbed in the next LayerNorm within the next residual stream. This allows us to copy the entire first part of the post-LN proof by mimicking the trajectories of the unit ball, while adding the constant amount of $\frac{\alpha_{ki}^l m\sqrt{d}}{2\sqrt{d+m^2/4}}$—multiple of all-ones vector in each round. Therefore, after the first N blocks, the projections of all the samples on the zero-sum hyperplane are identical to those in the post-LN proof. Each sample, however, has a different component in the direction of the all-one vector. This will, however, be absorbed by the last LayerNorm. Moreover, by triangle inequality, the margin of the trajectories in this extended space is at least as big as the margin of the trajectories on the zero-mean ball. The construction for the next $\tilde{K}$ blocks works by the same reasoning as well. Thus, after these layers, the projections of the samples on the zero-sum ball are identical to the post-LN proof and the last LayerNorm will absorb the component along the all-ones vector.

As for the transformers, although after the first block the samples are not centered and do not all have norm $\sqrt{d}$, after applying the LayerNorm in the first subsequent MLP block, they will all be distinct (except the ones with identical contexts). Therefore, we define the same trajectories as in the ResNet construction with the centered and normalized features, but we will perform an equivalent movement on the zero-mean ball with the radius equal to the norm of the projection of the particular sample onto the zero-mean hyperplane, while ignoring the all-ones component completely. As a result, each sample moves on its own cylinder, a projection to the zero-mean hyperplane following the trajectories on the normalized zero-mean ball, while moving arbitrarily along the all-ones direction. As before, a triangle inequality guarantees that the margin defined on the zero-mean normalized ball is not violated in the wider space during this process. The only caveat is that, if the norm of the ball along which a sample is traveling is larger than that of the $\sqrt{d}$-normed ball, we need to upscale α_{ki}^l by that ratio. Note that the size of the vector after the first block is upper bounded independently of the number of layers, therefore such an upscaling will only multiply the cost of weight matrices by a constant. The rest of the argument follows that of the adaptation for pre-LN ResNets.

Appendix H. Two linear layers per block with non-vanishing or uniform weight decay

Here, we intuitively describe why the NC metrics in general *do not* approach the perfect NC in architectures with two linear layers per residual block as the depth goes to infinity, *if the regularization is non-vanishing or vanishes uniformly across all layers*. The key is the simple inequality $\|AB\|_F \leq \|A\|_F \|B\|_F$. We can interpret these matrices as features, weight matrices and the change on the features added to the residual. In particular, in a ResNet with a single linear layer per block we have $\|\Delta X_l\|_F \leq \|W_l\|_F \|X_l\|_F$ (ΔX_l is the outcome of the residual branch added back to residual stream) and, importantly, this inequality can be made equality in some cases. Even if the inequality does not hold as an equality, we still have that, for fixed W_l, X_l, if $\Delta X_l = \sigma(W_l X_l) \neq 0$, then due to homogeneity $c\Delta X_l = (cW_l)X_l$. This makes the total change W_l makes to X_l scale linearly with c, but its cost is quadratic. Therefore, if the directional derivative of the loss w.r.t. ΔX_l at layer l is strictly positive, then there exists $c > 0$ for which cW_l *will* make an improvement against $W_l = 0$. However, if two linear layers are involved, we have

$$\|\Delta X_l\|_F \leq \|W_{l,2}\sigma(W_{l,1}X_l)\|_F \leq \sqrt{N/4}\left(\|W_{l,1}\|_F^2 + \|W_{l,2}\|_F^2\right).$$

Therefore, any change to the features will scale *linearly with the regularization cost* of the matrices that were responsible for this change. In this case, the opposite to the previous statement holds: if the directional derivative of the loss w.r.t. ΔX_l is *small enough,* then for small c, the c-scaling of weight matrices will necessarily *worsen* the loss compared to doing nothing.

As we have seen in the proof of Lemma 9 for the MSE loss (but this also holds for the CE loss), an $\mathcal{O}(\epsilon)$-sized perturbation around the global optimum of neural collapse causes only an $\mathcal{O}(\epsilon^2)$ increase in the loss. Furthermore, the derivative is zero at NC and locally Lipschitz around that point, which implies that the size of the derivative is $\mathcal{O}(\epsilon)$. For any input dataset X that is not yet collapsed, if the points W_L, X_L in the set of global optima $\tilde{\mathcal{M}}_{L,2}$ *did* approach NC in the limit, we could, by contradiction, take an optimum that is ϵ-close to NC (for a sufficiently small ϵ) and zero-out all the last layers that were responsible for moving the samples by a total amount of $\Theta(\epsilon)$ shift (this would need care in a rigorous proof because of the possible discontinuity of the layer-to-feature mapping). The change in the fit part of the loss would be $\mathcal{O}(\epsilon^2)$, but thanks to the above inequality, the total regularization cost saved by this would be $\Omega(\epsilon)$, so the loss would improve and we would arrive at a contradiction.

The above argument holds for constant regularization λ. However, even if the regularization was vanishing, but it was the same for W_L and for the rest of the network, the NC would still not be approached. To see this for MSE loss, consider a perturbed perfect scenario where the input data is $X = I_K \otimes \mathbf{1}_n^T + E$ and E is a perturbation matrix of size $\Theta(\epsilon)$. X is already ϵ-close to NC. To move X $\Theta(\epsilon)$ closer to NC, we need $\Theta(\lambda_L \epsilon)$ cost in terms of the weight matrices. Let us now compute the improvement in the corresponding GUFM objective that results from doing so. The DUFM objective with MSE is $\frac{1}{2N}\|WX_L - Y\|_F^2 + \frac{\lambda_L}{2}\|W\|_F^2$. If we simplify the problem to just fitting a single row of W (the optimization problem is separable, so this is w.l.o.g.), we have a simple ridge regression solution for w. In particular

$$w^* = (nI_K + \lambda_L I_K + \mathcal{O}(\epsilon))^{-1}(I_K \otimes \mathbf{1}_n^T + \mathcal{O}(\epsilon))y.$$

Therefore, the distance from the unperturbed fit $(nI_K + \lambda_L I_K)^{-1}(I_K \otimes \mathbf{1}_n^T)y$ is itself $\mathcal{O}(\epsilon)$ and plugging this in the loss, we see that the change in the loss function is $\mathcal{O}(\lambda_L \epsilon^2)$ which, for sufficiently small ϵ, is less than the price in terms of weight regularization.

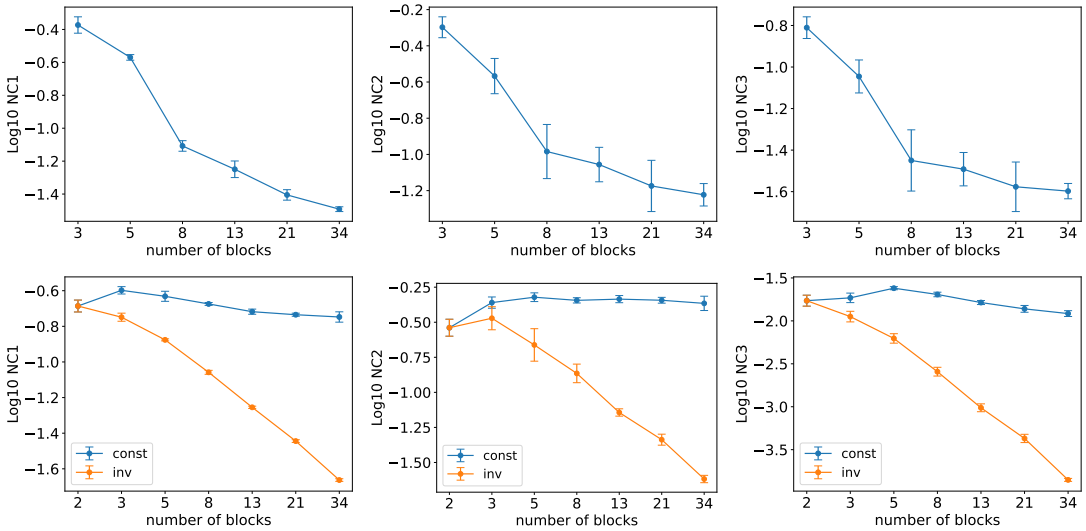

Figure 2: $\log_{10}$ of NC1, NC2 and NC3 metrics respectively in the left, middle and right column, as a function of L. *Top:* pre-LN L-T11 on IMDB; *Bottom:* L-RN2 on MNIST with constant $\lambda = 0.0025$ and $\lambda = 0.005L^{-1}$.

Appendix I. Additional experimental results and details

We train ResNets and transformers on MNIST [28], CIFAR10 [30] and IMDB [20] with increasing depths in $\{2, 3, 5, 8, 13, 21, 34\}$. The hidden dimension is 64, the learning rate 0.005 for vision and 0.001 for language and the regularization 0.005 for architectures having one linear layer per block and $0.005/L$ for architectures having two linear layers per block. Each setting is trained for 5 different random seeds for 5000 epochs on CE loss, the results are averaged, and the error bars at one standard deviation are reported. We use pre-LN transformers for language experiments and, due to training instabilities, only report the runs which converged by the end of the training.

In Figure 2, we provide additional experimental results. In particular, the top row considers the pre-LN L-T11 trained on the IMDB dataset, while the bottom row considers the L-RN2 with both constant 0.0025 and variable $0.005/L$ weight decay. The results and the message are consistent with those of Section 5. We can also see that constant regularization in L-RN2 does not bring almost any improvement in NC metrics or even worsens them, in contrast with scaling the weight decay as $1/L$.